

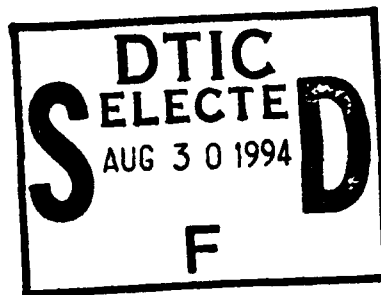
AD-A283 874

ADST/TR 94-003258



1

**ADST ARWA  
VISUAL SYSTEM MODULE  
SOFTWARE DESIGN DOCUMENT**



Loral Advanced Distributed Simulation  
12151-A Research Parkway  
Orlando, FL 32826-3283

*Date : 04-MAR-1994*

Contract No. N61339-91-D-0001  
Delivery Order No. 0048  
CDRL A004

This document has been approved  
for public release and sale; its  
distribution is unlimited.

Prepared for

Simulation Training and Instrumentation Command  
12350 Research Parkway  
Orlando, FL 32826-3275

**LORAL**  
Advanced Distributed Simulation

94-27971



12/97

DTIC QUALITY INSPECTED 5

94 8 29 261

August 1, 1994

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form approved OMB No. 0704-0</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE 28 February 1994		3. REPORT TYPE AND DATES COVERED Version 1
4. TITLE AND SUBTITLE ADST ARWA Visual System Module Software Design Document			5. FUNDING NUMBERS Contract No. N61339-91 Delivery Order Number 004 CDRL Number A004	
6. AUTHOR(S) Robert Anschuetz				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Loral Systems Company ADST Program Office 12151-A Research Parkway Orlando, FL 32826			8. PERFORMING ORGANIZATION REPORT NUMBER ADST/WDL/TR-94-00325	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Simulation, Training and Instrumentation Command STRICOM Naval Training Systems Center 12350 Research Parkway Orlando, FL 32826-3275			10. SPONSORING ORGANIZATION REPORT NUMBER A004	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE  A	
13. ABSTRACT (Maximum 200 words) The ADST ARWA Visual System Module Software Design Document describes the system design of the VSM CSCI. This Software Design Document outlines the structure and composition of the CSCI sub-funcs (CSCs and CSUs) and provides a detailed description of each.				
14. SUBJECT TERMS			15. NUMBER OF PAGES	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED		17. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED		17. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED
				20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)  
Prescribed by ANSI Std Z39-18  
298-102

1 SCOPE .....	1
1.1 Identification.....	1
1.2 System Overview .....	1
1.3 Document Overview .....	1
2 REFERENCED DOCUMENTS.....	1
3 PRELIMINARY DESIGN .....	2
3.1 CSCI Overview .....	2
3.1.1 CSCI Architecture .....	5
3.1.2 System States and Modes.....	8
3.1.3 Memory and Processing Time Allocation.....	12
3.2 CSCI Design Description .....	14
3.2.1 VSM Network Interface .....	21
3.2.1.1 BIU Decoder.....	21
3.2.1.2 VSM Language Decoder .....	21
3.2.1.3 VSM Language Encoder.....	21
3.2.1.4 BIU Encoder .....	21
3.2.2 VSM User Interface.....	22
3.2.2.1 Control Language Interpreter .....	22
3.2.2.2 Information Display Manager.....	22
3.2.2.3 Information Gathering Manager.....	22
3.2.3 VSM Hardware Interface .....	22
3.2.3.1 Joystick Interface .....	23
3.2.3.2 CIG Interface.....	23
3.2.3.3 HMD Interface.....	23
3.2.3.4 OTW Displays Interface .....	23
3.2.3.5 HT Interface .....	24
3.2.3.5.1 HT Hardware Interface .....	24
3.2.3.5.2 Head Position Prediction .....	24
3.2.3.5.3 Head Position Correction.....	24
3.2.4 VSM Resource Manager.....	24
3.2.4.1 HMD Manager .....	25
3.2.4.2 Status Manager.....	25
3.2.4.3 Head Tracker Manager .....	25
3.2.4.4 OTW Manager.....	25
3.2.4.5 Joystick Manager.....	26
3.2.4.6 Interface Manager .....	26
3.2.4.7 CIG Resources Manager.....	26
3.2.4.7.1 Ownship Manager .....	27
3.2.4.7.2 Environment Effects Manager.....	27
3.2.4.7.3 Special Effects Manager .....	28
3.2.4.7.4 Scene Content Manager .....	28
3.2.4.7.5 Data Base Manager .....	29
3.2.4.7.6 Moving Model Manager .....	29
3.2.4.7.7 Visual Channel Manager .....	30
3.2.5 Process Scheduler.....	30
4 DETAILED DESIGN.....	31

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

5 CSCI DATA.....	31
6 CSCI DATA FILES.....	31
7 REQUIREMENTS TRACEABILITY .....	31
8 NOTES .....	39
9 APPENDICIES.....	39

## LIST OF FIGURES

Figure 1. VSC, dfd Context-Diagram .....	4
Figure 2. VSC, dfd 0.....	7
Figure 3. VSC, Modes and States .....	11
Figure 4. VSC, Memory and Processing Time .....	13
Figure 5. VSC, dfd 1 .....	15
Figure 6. VSC, dfd 2.....	16
Figure 7. VSC, dfd 3.....	17
Figure 8. VSC, dfd 3.4.....	18
Figure 9. VSC, dfd 4.....	19
Figure 10. VSC, dfd 4.2.....	20

## **1 SCOPE**

The scope of this Software Design Document (SDD) is discussed in the following subparagraphs 1.1, 1.2, and 1.3.

### **1.1 Identification**

This document applies to the Advanced Rotary Wing Aircraft (ARWA) Visual System Module (VSM) Computer Software Configuration Item (CSCI). The requirements from which the design of this CSCI were derived are from the ARWA VSM System/Segment Specification (SSS).

### **1.2 System Overview**

The principal purpose of the Visual System Module is to simulate out-the-window (OTW) and sensor imagery and to display the imagery to the crew members of an ARWA device.

### **1.3 Document Overview**

The purpose of this document is to describe the system design of the ARWA VSM CSCI. This Software Design Document outlines the structure and composition of the CSCI sub-functions (Computer Software Components (CSCs) and Computer Software Units (CSUs)) and provides a detailed description of each. Sections 1, 2, 3, 7, and 8 are produced during preliminary design. Sections 4, 5, and 6 are produced during detailed design.

Section 1 outlines the scope of the document.

Section 2 describes the documents referenced in this specification.

Section 3 outlines preliminary design overview of the CSCI and a description of each CSC's purpose.

Section 4 describes the detailed design of each CSC.

Section 5 provides a detailed breakdown of all data elements defined for each CSC and for each associated CSU.

Section 6 provides a data file cross reference to aid in locating functions.

Section 7 provides a requirements traceability matrix.

Section 8 provides general design notes.

## **2 REFERENCED DOCUMENTS**

The following documents of the exact issue shown form a part of this specification to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this specification, the contents of this specification shall be considered a superseding requirement.

Non-Government

System/Segment Specification for the Advanced Rotary Wing Aircraft Visual System Module.

### 3 PRELIMINARY DESIGN

The following subparagraphs describe the preliminary design of the Visual System Module CSCI.

#### 3.1 CSCI Overview

The principal purpose of the VSM is to simulate out-the- window and sensor visual imagery and display this imagery to the crewmembers in the ARWA device. Imagery will be sufficient to allow replication of visual related piloting and weapons/sensor tasks for simulated warfighting exercises on the Distributed Interactive Simulation (DIS) network.

The Visual System Controller (VSC) is the software portion of the VSM that controls the display of images on the simulator displays in real time. The VSC communicates with the ARWA Global network, receiving information about ownship position, threats, weapons, environment, sensors, and other visual related information. The VSC processes this information and reformats it into buffers which can be understood by the Computer Image Generator (CIG). The information in these buffers is then passed to the CIG as updates to the current ownship status and processed by the CIG, resulting in a new visual scene. The VSC also monitors peripheral devices such as Out-The-Window Display System and the Head Tracker System and sends information about the system status and error conditions to the ARWA Global network.

The interfaces through which the VSC communicates with external entities are summarized below:

Identifier: HT\_To\_VSC

Name: Head Tracker to VSC

Description: Head tracker position and status information

Identifier: VSC\_To\_HT

Name: VSC to Head Tracker

Description: Initialization and setup commands

Identifier: HMD\_To\_VSC

Name: Helmet Mounted Display to VSC

Description: Helmet Mounted Display status information

Identifier: VSC\_To\_HMD

Name: VSC to Helmet Mounted Display

Description: Initialization and setup commands

Identifier: Joystick\_To\_VSC

Name: Joystick to VSC

**Description: Joystick position and status information**

**Identifier: VSC\_To\_Joystick**

**Name: VSC to Joystick**

**Description: Initialization and setup commands**

**Identifier: CIG\_To\_VSC**

**Name: Computer Image Generator to VSC**

**Description: Computer Image Generator status information**

**Identifier: VSC\_To\_CIG**

**Name: VSC to Computer Image Generator**

**Description: Initialization and control commands**

**Identifier: Console\_To\_VSC**

**Name: Administrative Console to VSC**

**Description: Administrative Console status and keyboard information**

**Identifier: VSC\_To\_Console**

**Name: VSC to Administrative Console**

**Description: Screen initialization and control commands**

**Identifier: FDDI\_Global\_Bus\_To\_VSC**

**Name: FDDI Global Bus to VSC**

**Description: FDDI Global Bus status and network information obtained from the SSM and FSM**

**Identifier: VSC\_To\_FDDI\_Global\_Bus**

**Name: VSC to FDDI Global Bus**

**Description: Network information and status destined for the SSM and FSM**

**Identifier: OTW\_Displays\_To\_VSC**

**Name: Out-The-Window Displays to VSC**

**Description: Out-The-Window Displays status information**

**Identifier: VSC\_To\_OTW\_Displays**

**Name: VSC to Out-The-Window Displays**

**Description: Status request**

**The following figure depicts the Context Diagram of the Visual System Controller, including external entities and external interfaces.**

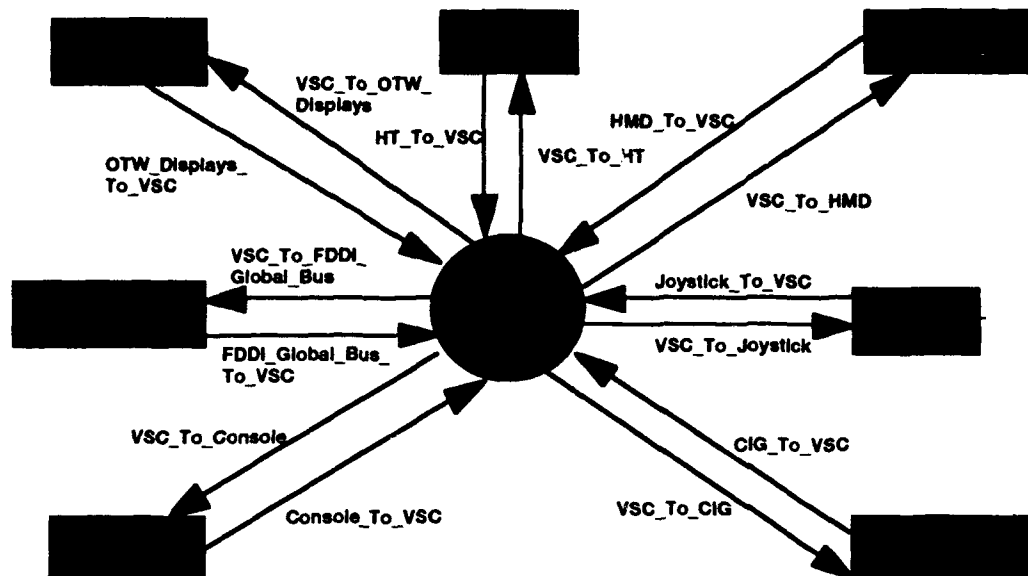


Figure 1. VSC, dfd Context-Diagram



### 3.1.1 CSCI Architecture

The Visual System Controller has been broken down into 5 top-level CSCs as follows:

CSC Identifier: VSM\_Network\_Interface

CSC Purpose: Provides an interface from the Fiber Distributed Data Interface (FDDI) Global Bus to the VSC for communication between the VSM and the Flight Station Module (FSM) and Simulation System Module (SSM) CSCIs.

CSC Identifier: VSM\_User\_Interface

CSC Purpose: Provides an interface from the Administrative Console to the VSC for communication between the VSM and the operator in stand-alone mode.

CSC Identifier: VSM\_Hardware\_Interface

CSC Purpose: Provides an interface from external hardware devices to the VSC for communication between the VSM and the CIG, Head Tracker, Helmet Mounted Display, Out-The-Window Displays, and Joystick.

CSC Identifier: VSM\_Resource\_Manager

CSC Purpose: Provides internal processing of VSM functions.

CSC Identifier: Process\_Scheduler

CSC Purpose: Provides scheduling and invocations of VSM functions.

The VSC CSC-to-CSC interfaces are described as follows:

Interface: General\_Stat

Name: General Statistics

Description: Hardware and Processing Statistics

Interface: Net\_Interface\_Control

Name: Network Interface Control

Description: FDDI Global Bus Commands from SSM and FSM

Interface: Net\_Interface\_Data

Name: Network Interface Data

Description: FDDI Global Bus Data to SSM and FSM

Interface: User\_Interface\_Control

Name: User Interface Control

Description: Administrative Console Commands

Interface: User\_Interface\_Data

Name: User Interface Data

Description: Administrative Console Screen Information

Interface: Hardware\_Interface\_Data

Name: Hardware Interface Data

Description: Hardware Data from CIG, Head Tracker, Helmet Mounted Display, Out-The-Window Display, and Joystick

Interface: Hardware\_Interface\_Control

Name: Hardware Interface Control

Description: Hardware Control going to CIG, Head Tracker, Helmet Mounted Display, Out-The-Window Display, and Joystick

Interface: Hardware\_Stat

Name: Hardware Status

Description: Hardware status obtained from the CIG, Head Tracker, Helmet Mounted Display, Out-The-Window Display, and Joystick

The following figure depicts the top-level Data Flow Diagram (DFD) for the Visual System Controller, illustrating the CSCI architecture.

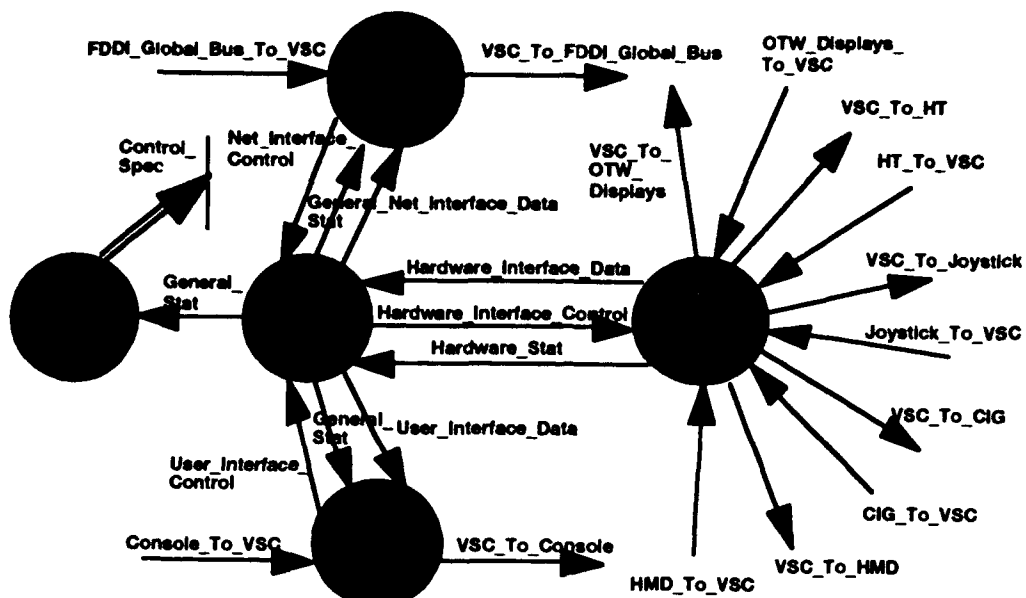


Figure 2. VSC, dfd 0

## 1.2 System States and Modes

The Visual System Controller shall operate within the various modes and states as defined hereafter.

### Module Mode.

This mode has two states: Start-up Initialization and Stand-alone.

#### Start-Up Initialization State.

Upon power up, each component of the VSC, as appropriate, shall execute internally stored boot-programs to initialize communications paths within the VSC and between the VSC and the ARWA Global Network. The VSC Function shall then direct the loading of the various initialization and diagnostic programs to initialize each VSC component to a state in which to receive simulation specific application programs, to perform morning readiness diagnostics, and to report pass/fail availability of the VSC to the Simulation Manager. Based upon commands from the Simulation Manager, the VSC then transitions to System Mode or enters the stand-alone state of Module Mode.

#### Stand-Alone State.

The initial transition to stand-alone mode is accompanied by the retrieval, loading, and execution of the VSC program to control the stand-alone state. In this state the VSC shall respond to controls and data which are received via the ARWA Global Network. These controls and data can originate with the Simulation Manager or with any user station on the network.

#### System Mode.

The initial transition to this mode is accompanied by the retrieval, loading and execution of any programs required to establish the VSC and ARWA Global Network interface. In this mode the VSC awaits mode transition commands on the ARWA Global Network. From the System Mode, the VSC can transition to the Remote Controlled Diagnostic Mode, the Simulation Mode, or the Shutdown Mode.

#### Simulation Mode.

This mode has four states: Initialization, Alignment, Total Freeze, and Run.

#### Initialization State.

Upon transition to Simulation Mode, the VSC is in this state. When entering this state from System Mode, the VSC shall load and execute the simulation programs for the various VSC components. When components are executing their programs, the module is ready to respond to a state transition command to enter the Alignment State or return to the System Mode.

#### Alignment State.

August 1, 1994

Upon entry into this state, the VSC shall retrieve the relevant data definition files from the Software and Data Base Maintenance stations. If the appropriate data base files are not resident in the VSC data base storage devices (the wrong or incorrect version of the data base is loaded), the VSC shall coordinate the loading of the appropriate data bases. The VSC shall then proceed to initialize the other VSC components as appropriate to the conditions specified in the data definition files. These conditions shall include the terrain data base position of the ownship and the initial modes of the sensors. Therefore, the terrain data appropriate for the position of the ownship can be retrieved and loaded into the Computer Image Generator. When all data loading and other VSC initialization is completed, a completion message is communicated over the ARWA Global Network and the VSC stands ready for a transition command to the Total Freeze State.

#### Total Freeze State.

In this state the components of the VSC are expecting the simulation programs; however, no internal parameters, such as ownship position, nor other aspects of image content, are being updated.

#### Run State.

In this state the VSC is producing images based on control parameters received via the ARWA Global bus and from the Head Tracker.

#### Remote Controlled Diagnostic Mode.

The Visual Module shall support this mode by responding as appropriate to the two self-test commands which are received via the ARWA Global Network.

#### Shut Down Mode.

The VSC has no required actions in this mode. When the command to transition into Shut Down Mode from System Mode is received, the VSC shall transition into Module Mode.

[illegible]

August 1, 1994

USER_INTERFACE_DATA	VSM_RESOURCE_MANAGER	VSM_USER_INTERFACE
HARDWARE_INTERFACE_CONTROL	VSM_RESOURCE_MANAGER	VSM_HARDWARE_INTERFACE
HARDWARE_STAT	VSM_HARDWARE_INTERFACE	VSM_RESOURCE_MANAGER
HARDWARE_INTERFACE_DATA	VSM_HARDWARE_INTERFACE	VSM_RESOURCE_MANAGER

**Figure 3. Modes and States**

### 3.1.3 Memory and Processing Time Allocation

The following table illustrates the memory and processing time allocations for the top level CSCs of the VSM.



CSC Name	Memory Budget	Allocated Processing Time
VSM_NETWORK_INTERFACE	2,000 Lines of Code	1 Millisecond
VSM_USER_INTERFACE	6,000 Lines of Code	3 Milliseconds
VSM_HARDWARE_INTERFACE	10,000 Lines of Code	5 Milliseconds
VSM_RESOURCE_MANAGER	44,000 Lines of Code	20 Milliseconds
PROCESS_SCHEDULER	2,000 Lines of Code	1 Millisecond
TOTAL	1 Megabyte	30 Milliseconds
AVAILABLE	48 Megabytes	33.3 Milliseconds
RESERVE	47 Megabytes	3.3 Milliseconds
RESERVE PERCENT	98%	10%

**Figure 4. Memory and Processing Time**

### 3.2 CSCI Design Description

The following paragraphs describe the top-level and lower-level CSCs of the Visual System Controller. The following figures depict the Data Flow Diagram breakdown of the Visual System Controller.

August 1, 1994

ADST/TR 94-003258

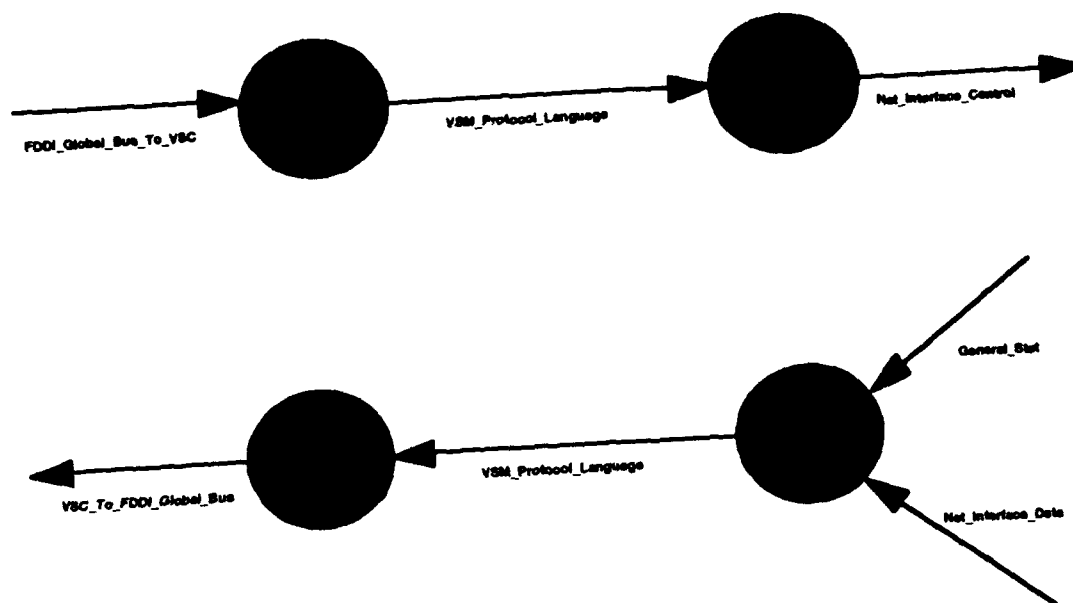


Figure 5. VSC, dfd 1

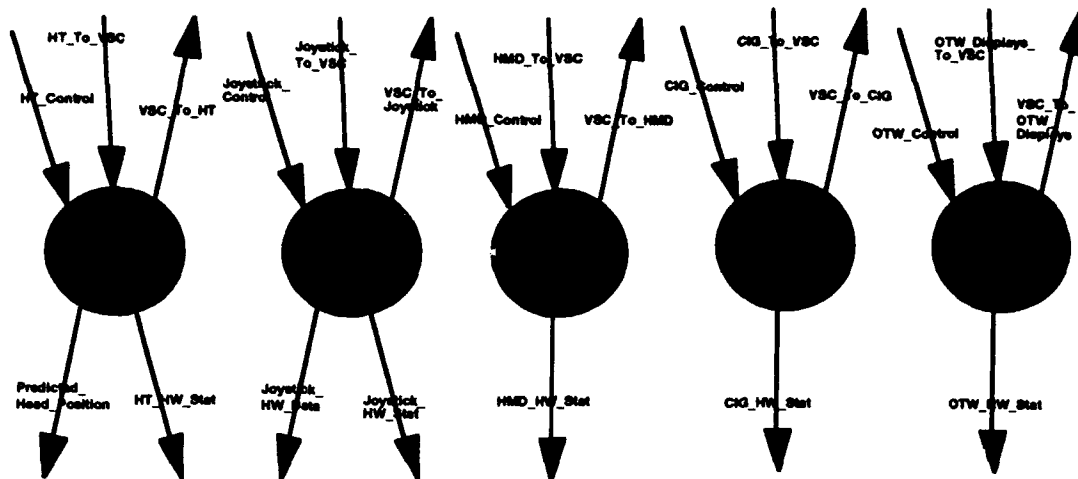


Figure 7. VSC, dfd 3

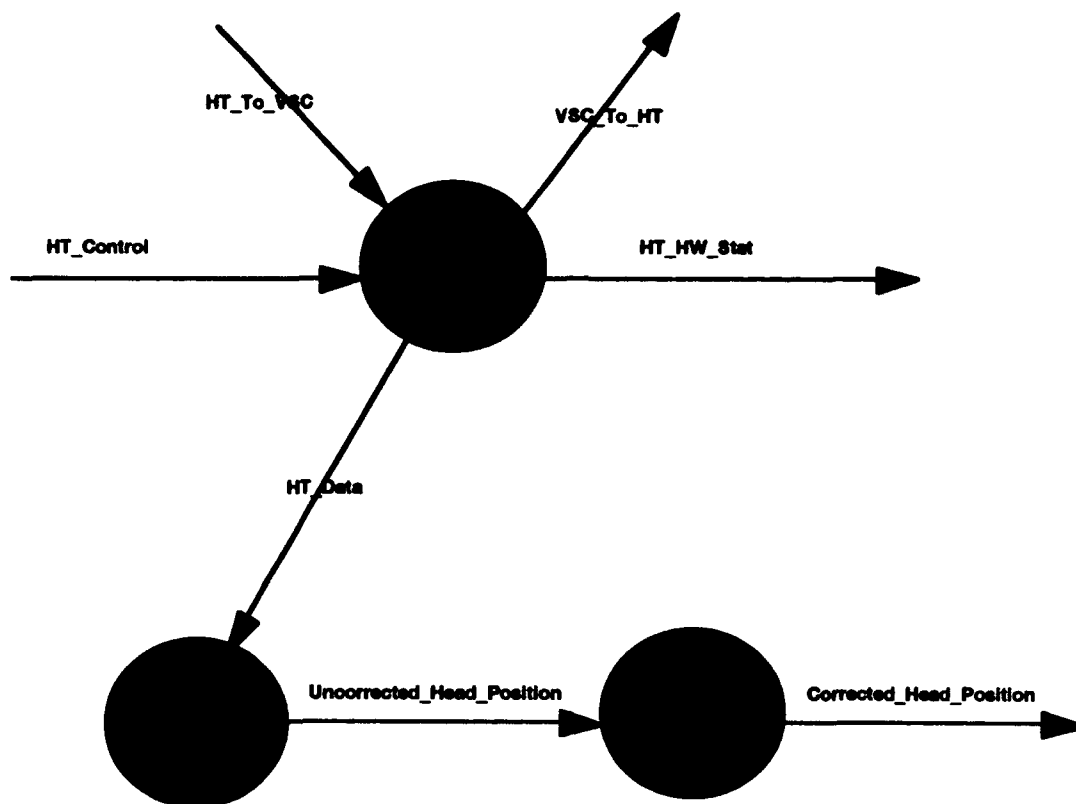


Figure 8. VSC, dfd 3.4

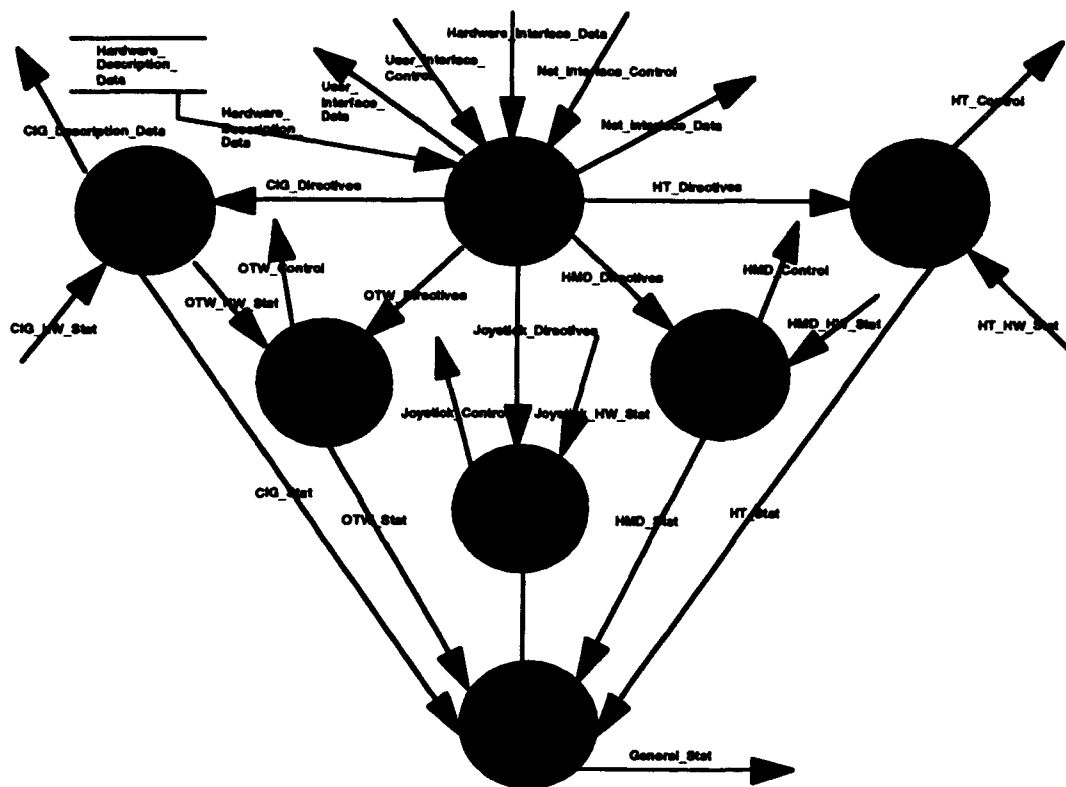


Figure 9. VSC, dfd 4

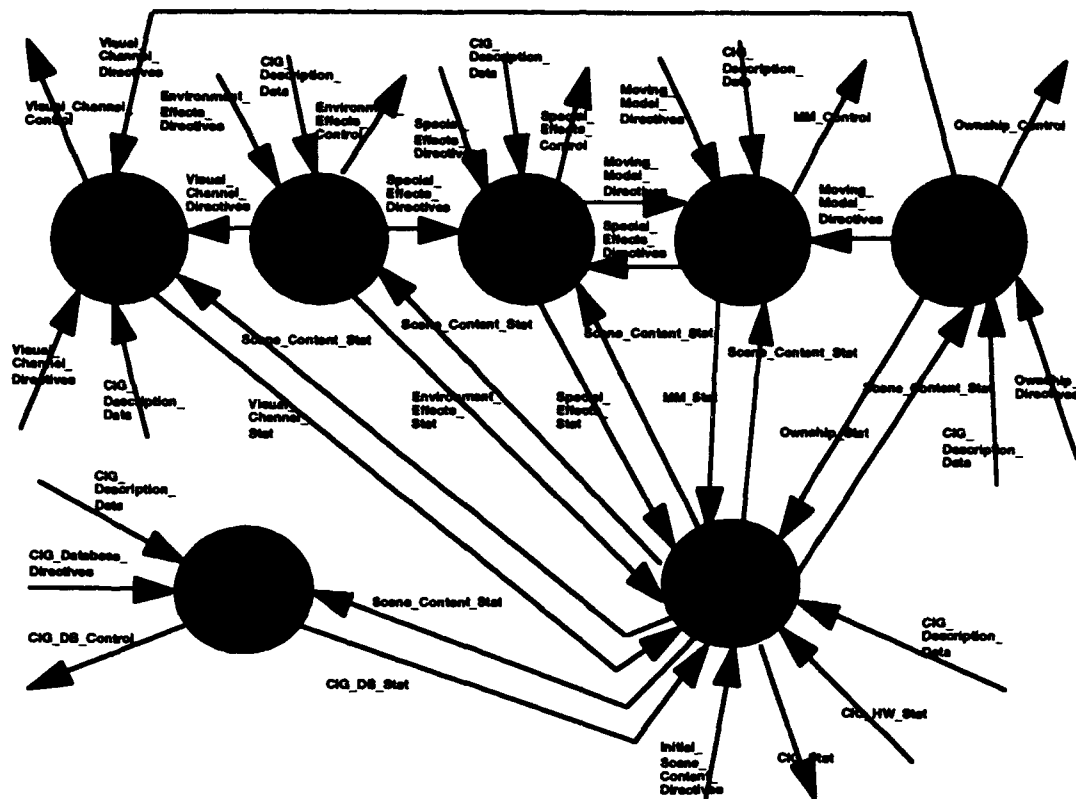


Figure 10. VSC, dfd 4.2

### **3.2.1 VSM Network Interface**

#### **VSM\_Network\_Interface**

- VSM\_Network\_Interface is composed of the following processes:

BIU\_Decoder  
VSM\_Language\_Decoder  
VSM\_Language\_Encoder  
BIU\_Encoder

#### **3.2.1.1 BIU Decoder**

##### **BIU\_Decoder**

- Converts FDDI global bus BIU protocol to VSM protocol.
- Checks messages on FDDI global bus for messages having a VSM destination.
- Communicates to hardware through Bus Interface Unit (BIU) layered structure.
- Communicates via Transmission Communication Protocol / Internet Protocol (TCP/IP) over FDDI.

#### **3.2.1.2 VSM Language Decoder**

##### **VSM\_Language\_Decoder**

- Accepts VSM protocol language and translates it to commands destined for the CIG, out-the-window display, joystick, helmet mounted display, and head tracker managers.

#### **3.2.1.3 VSM Language Encoder**

##### **VSM\_Language\_Encoder**

- Accepts data and statistics destined for the SSM and FSM.
- Translates interface data and general statistics to VSM protocol language.

#### **3.2.1.4 BIU Encoder**

##### **BIU\_Encoder**

- Converts VSM protocol to FDDI global bus BIU protocol.
- Attaches destination header to message to send to correct node.



- Communicates to hardware through BIU layered structure.
- Communicates via TCP/IP over FDDI.

### **3.2.2 VSM User Interface**

#### **VSM\_User\_Interface**

- VSM\_User\_Interface is composed of the following processes:

Control\_Language\_Interpreter  
Information\_Display\_Manager  
Information\_Gathering\_Manager

#### **3.2.2.1 Control Language Interpreter**

##### **Control\_Language\_Interpreter**

- Accepts keyboard and pointer device selection from the administrative console.
- Communicates via RS-232.
- Formats RS-232 inputs into CIG command line interpreter commands.

#### **3.2.2.2 Information Display Manager**

##### **Information\_Display\_Manager**

- Receives display information destined for output on the console terminal.
- Has display routines for displaying information.
- Communicates via RS-232.

#### **3.2.2.3 Information Gathering Manager**

##### **Information\_Gathering\_Manager**

- Accepts user interface data and general statistics.
- Returns user interface control information, which consists of Command Line Interpreter (CLI) commands.
- Formats display information based on user interface data and general statistics.

### **3.2.3 VSM Hardware Interface**

## **VSM\_Hardware\_Interface**

- VSM\_Hardware\_Interface is composed of the following processes:

Joystick\_Interface  
CIG\_Interface  
HMD\_Interface  
OTW\_Displays\_Interface  
HT\_Interface

### **3.2.3.1 Joystick Interface**

#### **Joystick\_Interface**

- Communicates via RS-232.
- Accepts data and converts to device independent data and status.
- Accepts joystick commands and converts to hardware specific device commands.

### **3.2.3.2 CIG Interface**

#### **CIG\_Interface**

- Communicates via ScramNet.
- Takes ScramNet data and converts to device independent status.
- Accepts CIG commands and converts to hardware specific device commands.

### **3.2.3.3 HMD Interface**

#### **HMD\_Interface**

- Accepts helmet mounted display commands and converts them to hardware specific device commands.
- Takes interface data and converts to a device independent status.
- Communicates via a serial interface.

### **3.2.3.4 OTW Displays Interface**

#### **OTW\_Displays\_Interface**

- Communicates via a serial interface.
- Takes hardware data and converts to device independent status.
- Accepts out-the-window status check command and converts to a hardware specific device command.

### **3.2.3.5 HT Interface**

#### **HT\_Interface**

- HT\_Interface is composed of the following processes:

HT\_Hardware\_Interface  
Head\_Position\_Prediction  
Head\_Position\_Correction

#### **3.2.3.5.1 HT Hardware Interface**

##### **HT\_Hardware\_Interface**

- Communicates with head tracking device through a serial protocol.
- Accepts head tracker control commands and translates them into hardware commands for the head tracker.
- Receives head tracker inputs and translates into hardware independent head tracker data and status.

#### **3.2.3.5.2 Head Position Prediction**

##### **Head\_Position\_Prediction**

- Receives head tracker position data and predicts head tracker position based on velocity, acceleration, and previous position.
- Performs damping and filtering functions.
- Sends out predicted head position to correction algorithm.

#### **3.2.3.5.3 Head Position Correction**

##### **Head\_Position\_Correction**

- Receives predicted head position data and corrects for multiple eyepoints.
- Sends out predicted head position.

### **3.2.4 VSM Resource Manager**

#### **VSM\_Resource\_Manager**

- VSM\_Resource\_Manager is composed of the following processes:

HMD\_Manager  
Status\_Manager  
Head\_Tracker\_Manager  
OTW\_Manager  
Joystick\_Manager  
Interface\_Manager  
CIG\_Resources\_Manager

#### **3.2.4.1 HMD Manager**

##### **HMD\_Manager**

- Accepts helmet mounted display hardware status and helmet mounted display directives.
- Passes on helmet mounted display status to the status manager.
- Translates helmet mounted display directives to helmet mounted display control commands.

#### **3.2.4.2 Status Manager**

##### **Status\_Manager**

- Accumulates status from CIG, out-the-window display, joystick, helmet mounted display, and head tracker managers.
- Sends out general status summarizing the state of the VSM.

#### **3.2.4.3 Head Tracker Manager**

##### **Head\_Tracker\_Manager**

- Accepts head tracker hardware status and head tracker directives.
- Passes on head tracker status to the status manager.
- Translates head tracker directives to head tracker control commands.

#### **3.2.4.4 OTW Manager**

**OTW\_Manager**

- Accepts out-the-window hardware status and out-the-window directives.
- Passes out-the-window status to the status manager.
- Translates out-the-window status check directive to out-the-window status check control command.

**2.4.5 Joystick Manager****Joystick\_Manager**

Accepts joystick hardware status and joystick directives.

Passes on joystick status to the status manager.

Translates joystick directives to joystick control commands.

**2.4.6 Interface Manager****Interface\_Manager**

Accepts hardware interface data, user interface control, and network interface control.

Converts inputs into directives to the CIG, out-the-window display, joystick, helmet mounted display, and head tracker.

Redirects any necessary user interface data or network interface data from hardware interface data, specifically head tracker data.

Reads the hardware description data file, which affects how directives are processed.

In stand-alone mode, converts head tracker data to control directives.

**2.4.7 CIG Resources Manager****CIG\_Resources\_Manager**

CIG\_Resources\_Manager is composed of the following processes:

Friendship\_Manager  
 Environment\_Effects\_Manager  
 Special\_Effects\_Manager  
 Scene\_Content\_Manager  
 Data\_Base\_Manager  
 Moving\_Model\_Manager

## Visual\_Channel\_Manager

### 3.2.4.7.1 Ownship Manager

#### Ownship\_Manager

Maps ownship directives to IG description data. Maps the type of vehicle for slewing and offsetting eyepoint.

Controls activating and deactivating ownship model.

Ownship configuration determines which aircraft is active.

Eyepoints are slewable for sensors, so require 3 eyepoints.

Sends a visual channel command for slewing to the visual channel manager.

Sends a definition for ownship moving model to the moving model manager. Therefore, most of the loading statistics for the ownship will be handled in the moving models area.

Ownship directives will include initial resolution and initial field of view.

Keeps track of loading history based on statistics obtained from the scene content manager.

Ownship entity data is treated in moving models. VSM needs to know the ID for ownship moving models, then treat the ownship as a moving model, except for items necessary for viewport changes.

### 3.2.4.7.2 Environment Effects Manager

#### Environment\_Effects\_Manager

SSM directives for rain, snow, time-of-day, cloud cover, storms, lightning, visibility, fog, haze, ground fog, wind, sun position are mapped using CIG description file.

Horizon effects for dawn and dusk are also mapped.

Environmental effects will allow for parameters which are not yet implemented. Opcode for cloud will include such information as type of cloud, layers, storms, lightning, ground to cloud, cloud to cloud, and any other pertinent information.

Lightning effects, lights for airports and cities, flashing beacons, and color of lights are also environmental effects.

Categorizes some items as special effects directives.

Generates visual control directives.

- Keeps track of how many environmental effects are in action currently.
- Prioritizes and weighs different effects for different CIGs.
- Keeps track of loading history based on statistics from the scene content manager.

#### 3.2.4.7.3 Special Effects Manager

##### Special\_Effects\_Manager

- Manages activation, deactivation, loading, cloning, unloading, and deleting of special effects sequences.
- Controls stop, start, pause, resume, and timed-run for special effects.
- Description data maps directive to a special effect for the CIG. May need to use moving models to perform special effects.
- Manages special effects based on range and time remaining. Has load prioritization table to determine which special effect would be eliminated when getting near overload conditions.
- Has internal list of active special effects in case a special effect leaves the field of view, so that the special effect can continue when the field of view returns.
- Based on IG information, needs to determine if a special effect can be started at any point in its sequence. Special effects may need to be activated even when outside the field of view. For example if a moving model is hit when it is outside the field of view, the special effect can begin in the middle of the sequence when it is brought to the field of view.
- Manages a list of special effects which are still active and keep timers for each one indicating how much time is left.
- Handles statistics about weights of complexity and passes back statistics about load management.
- There will not have external directive interfaces to this process. Either the moving model manager or the environmental model manager will decide when special effects are needed.
- Parameters may be added to the interface definition which allows modifications to the special effect.
- Keeps track of loading history based on statistics from scene content manager.

#### 3.2.4.7.4 Scene Content Manager

##### Scene\_Content\_Manager

- After other processes are done determining loading statistics, the scene content manager assesses the statistics and determines overload conditions. If an overload condition exists, it sends back statistics requesting new updating of active effects or models.
- When a manager sends a certain load to scene content manager, the scene content manager sends back a status. If overloaded, it will tell the managers how much percentage of the processing time is going to be allocated to each of the managers. The managers will send back once again how much of the percentage was used, and the scene content manager returns a good status.
- Initially sets percentage of processing power estimated for each manager. Then the percentage is dynamically changed based on prior knowledge when overloading is occurring. Managers send only amount allocated. Functions inside the scene content manager calculate percentages allocated for manager processing based on past history. Percentages will be smoothly reallocated.
- Gathers statistics from manager processes and passes on.

#### 3.2.4.7.5 Data Base Manager

##### Data\_Base\_Manager

- Activates, loads, and deactivates database by CIG database directives and CIG description data.
- Manages state of database, including level of detail, whether active, number of levels, and representation. Controls loading on system by changing parameters including blending, level-of-detail, field / range control, and feature representations.
- Controls culture (roads, lakes, rivers) and level of complexity. Controls level-of-detail load control based on range.
- Handles dynamic level-of-detail. Also handles blending and ranging control, plus continuous terrain transition range controls.
- Statistics determine percentage of processing time allocated.
- Keeps track of loading history based on statistics from scene content manager.
- Prepares the CIG for an overload condition before it occurs.

#### 3.2.4.7.6 Moving Model Manager

##### Moving\_Model\_Manager

- Interprets moving model directives and maps to the CIG description data.
- In stand-alone mode, if a directive comes from a console, invokes an internal flight model, which will either be a rotary wing model, a fixed wing model, or a simple movement model.



- Activates/deactivates, clones, deletes, changes characteristics, loads, unloads, and controls moving models.
- Creates a CIG control table for moving models.
- Controls articulated parts for moving models. Provides services for activating, binding and controlling data.
- Manages load control based on prioritization according to level of detail, complexity, percent occulted, and range. Determines loading criteria for each model in some fashion (such as the number of polygons per model).
- Deactivations will be handled by the VSM as they are received. Activations will be handled as soon as possible. VSM maintains queue of active models.
- Statistics and status will be passed to the scene content manager which describes loading and which moving models are active.
- Sends request to special effects manager to process moving models which require special effects.
- Determines overload conditions based on scene content statistics and hardware statistics.
- Keeps track of loading history based on statistics from the scene content manager.

#### 3.2.4.7.7 Visual Channel Manager

##### Visual\_Channel\_Manager

- Manages sensor and out-the-window channels. Manages each type of sensor.
- Interprets visual channel directives and maps to the CIG description data for such features as channel activation/deactivation, channel resolutions, fields of view, channel mixing, distortion correction, noise introduction, and sensor types.
- Creates instances of channels and configures channels based on ownership directives from the SSM. The out-the-window eyepoint and 2 sensor eyepoints are defined.
- Controls switching polarity and adding noise.
- Ownership manager sends directives to the visual channel manager which sends out commands for slewing.
- Keeps track of loading history based on statistics from scene content manager.

#### 3.2.5 Process Scheduler

## Process\_Scheduler

- Determines processes to be activated.
- Invokes processes based on priorities, hardware statistics, and dependencies.

## 4 DETAILED DESIGN

This section shall be completed during detailed design.

## 5 CSCI DATA

This section shall be completed during detailed design.

## 6 CSCI DATA FILES

This section shall be completed during detailed design.

## 7 REQUIREMENTS TRACEABILITY

The design outlined in this SDD can be traced back to System/Segment Specification for the ARWA Visual System Module. The following is a list of VSM software requirements which have been extracted from the VSM System/Segment Specification.

### VSM Requirements

#### 1. Generic (SSS 3.4.3)

Internal interfaces shall be unaffected by hardware changes. Software shall be designed to maximize hardware independence.

#### 2. Able to operate in stand-alone mode (SSS 3.1.3.1.2, 3.1.4.2.3.7)

By using the command language interpreter, joystick, head tracker position, and flight model (selectable between fixed-wing, rotary-wing, and inspection), VSM shall be able to operate without FDDI interfaces. Helmet mounted displays shall be operational, but no symbology shall be present.

#### 3. Parameter driven (SSS 3.1.3.3.2)

Configuration files shall drive the visual simulation so software shall not have to be recompiled whenever there is a configuration change. The configuration file shall be either a text or binary file, modified through a software configuration tool.

#### 4. Supports morning readiness (SSS 3.1.3.1.1, 3.1.4.2.3.1)

The visual system shall support the running of an internal hardware check where possible. Simple tests shall be developed for other devices. Rebooting the machine may supply status checks for readiness data. Primary concern shall be a pass/fail status for each hardware device.

**5. Supports diagnostics**

(SSS 3.1.3.4, 3.1.4.2.3.3)

Local mode diagnostics shall be supplied by hardware vendors whenever possible. Remote mode diagnostics shall be a pass/fail status check for each hardware device.

**6. Initialize mode**

(SSS 3.1.3.3.1, 3.1.4.2.3.2)

The VSC shall support initialization to a ready state. CIG startup commands may be driven from command files.

**7. Run mode**

(SSS 3.1.3.3.4)

The VSC shall support operational running.

**8. Exit mode**

(Derived)

The VSC shall support operational termination.

**9. Busy mode**

(Derived)

The VSC shall send back a busy state (during database load).

**10. Freeze mode**

(SSS 3.1.3.3.3, 3.1.4.2.3.4)

Partial freeze shall stop moving models, but continue to allow eyepoint to move. Total freeze shall freeze the CIG as well as VSC software. The only acceptable commands to the VSC in this state shall be mode change commands.

**11. Shutdown mode**

(SSS 3.1.3.5, 3.1.4.2.3.5)

Normal shutdown shall stop the process and clear hardware. Security shutdown shall stop accepting FDDI messages and transition to stand-alone mode

**12. Pause mode**

(Derived)

During pause mode a pause command shall be sent to the CIG.

**13. Send/receive from FDDI network**

(SSS 3.1.4.1)

The Bus Interface Unit shall control this activity and be SSM provided.

**14. Full CIG control**

(SSS 3.1.4.2.3.8)

The CIG shall be controlled within the boundaries of the defined VSM interface

#### 15. Moving models

(SSS 3.1.4.2.3.9.1, 3.1.4.3.3.10.2)

1000 moving models with up to 8 nested articulated parts shall be a design requirement which may be limited by hardware constraints. Prioritization of moving models shall be based on range, model priority, size, resolution, and location within the field of view.. 120 special effects shall be a design requirement which may be limited by hardware constraints. Prioritization of special effects shall be based on range, priority, size, resolution, timing, and location within the field of view. Requirements for moving models and special effects from the system/segment specification shall be as follows:

Moving Models   Special Effects	
Gaming Area	2000   240
Active Area	1000   120

#### 16. Sensor

(SSS 3.1.4.3.3.2, 3.1.4.3.3.6)

Field of view discrete changes shall be controlled through a continuous field of view interface definition. Field of view continuous changes shall be handled through the interface definition. If the CIG only supports discrete field of view changes, the closest approximation shall be chosen through the configuration file. Sensor type changes shall be mapped to the closest available display form through the configuration file. Sensor controls shall be available for gain, noise, and polarity.

#### 17. Load Management

(SSS 3.1.4.2.3.11)

Proactive load management techniques shall be employed. Learning techniques shall create a historical grid of data with loading parameters. Models shall also have histories. An example of a learning technique which might be employed is flying the entire database and loading all models to learn loading prior to simulator usage. The process scheduler shall learn the capabilities of the host processor. The following features shall be controlled moving models special effects terrain cultural features The following shall be possible methods of control activation deactivation blending level of detail frame rate

#### 18. Environmental control

(SSS 3.1.4.2.3.9.2, 3.1.4.3.3.4, 3.1.4.3.3.5, 3.1.4.3.3.10.3)

Time of day shall be continuous. The SSM shall supply the VSC with the initial time and day, and the VSC shall change the sun and moon position either with internal algorithms or CIG algorithms based on continuous time.

Light sources shall include sun, moon, stars, flares, and lightning.

Features include the following:

Clouds

Fog

Haze

density

color

atmospheric attenuation curves

Horizon band

Visibility

Scud

Thunderstorms

Smoke

#### 19. Process Control

(Derived)

The software shall provide the capability to disable certain CSCs (i.e. head tracker functions, load management, etc.) within the process scheduler. Battle damage may cause an alteration of process control.

#### 20. Transport delay

(SSS 3.1.4.3.3.3)

The VSC shall not add more than 33.3 milliseconds total from the receipt of data to the end of transmission

#### 21. Head tracking

(Derived)

Head tracking shall provide the following capabilities: Damping

Smoothness

Filtering

Prediction

Offset mapping

#### 22. Synchronization

(Derived)

The CIGs shall be synced according to the master CIG and according to the ability of the hardware.

#### 23. Channel definition and alignment

(Derived)

Programmable ownership definition shall meet this requirement.

#### 24. Joystick

(Derived)

Maintenance joystick shall be supported for stand-alone mode and testing.

The following requirements traceability list shows the allocation of VSM requirements to lower level VSM CSCs.

## VISUAL\_SYSTEM\_CONTROLLER REQUIREMENTS TRACEABILITY

## 1 VSM\_Network\_Interface CSC

## 1.1 BIU\_Decoder CSC

Requirements Satisfied:

Generic

Send/receive from FDDI network

## 1.2 VSM\_Language\_Decoder CSC

Requirements Satisfied:

Generic

## 1.3 VSM\_Language\_Encoder CSC

Requirements Satisfied:

Generic

## 1.4 BIU\_Encoder CSC

Requirements Satisfied:

Generic

Send/receive from FDDI network

## 2 VSM\_User\_Interface CSC

## 2.1 Control\_Language\_Interpreter CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

## 2.2 Information\_Display\_Manager CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

## 2.3 Information\_Gathering\_Manager CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

## 3 VSM\_Hardware\_Interface CSC

## 3.1 Joystick\_Interface CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Joystick

3.2 CIG\_Interface CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

3.3 OTW\_Displays\_Interface CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

3.4 HT\_Interface CSC

3.4.1 HT\_Hardware\_Interface CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Head tracking

3.4.2 Head\_Position\_Prediction CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Head tracking

3.4.3 Head\_Position\_Correction CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Head tracking

3.5 HMD\_Interface CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

4 VSM\_Resource\_Manager CSC

4.1 Status\_Manager CSC

Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Load Management

Supports morning readiness

Supports diagnostics

## 4.2 CIG\_Resource\_Manager CSC

### 4.2.1 Scene\_Content\_Manager CSC

#### Requirements Satisfied:

Generic Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

### 4.2.2 Ownship\_Manager CSC

#### Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

### 4.2.3 Moving\_Model\_Manager CSC

#### Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

Moving models

### 4.2.4 Special\_Effects\_Manager CSC

#### Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

Moving models

Environmental control

### 4.2.5 Environment\_Effects\_Manager CSC

#### Requirements Satisfied:

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

Environmental control

### 4.2.6 Data\_Base\_Manager CSC



**Requirements Satisfied:**

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

**4.2.7 Visual\_Channel\_Manager CSC**

**Requirements Satisfied:**

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

Full CIG control

Sensor

Channel definition and alignment

**4.3 Head\_Tracker\_Manager CSC**

**Requirements Satisfied:**

Generic

Able to operate in stand-alone mode

Load Management

Head tracking

**4.4 Interface\_Manager CSC**

**Requirements Satisfied:**

Generic

Able to operate in stand-alone mode

Load Management

Parameter driven

**4.5 Joystick\_Manager CSC**

**Requirements Satisfied:**

Generic

Able to operate in stand-alone mode

Load Management

Joystick

**4.6 OTW\_Display\_Manager CSC**

**Requirements Satisfied:**

Generic

Able to operate in stand-alone mode

Load Management

**4.7 HMD\_Manager CSC**

**Requirements Satisfied:**

**Generic**

Able to operate in stand-alone mode

Load Management

**5 Process\_Scheduler CSC**

Requirements Satisfied:

**Generic**

Able to operate in stand-alone mode

Initialize mode

Run mode

Exit mode

Busy mode

Freeze mode

Shutdown mode

Pause mode

Process control

Transport delay

Synchronization

**8 NOTES****ACRONYM LIST**

ARWA	Advance Rotary Wing Aircraft
BIU	Bus Interface Unit
CLI	Command Line Interpreter
CIG	Computer Image Generator
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
CSU	Computer Software Unit
DFD	Data Flow Diagram
DIS	Distributed Interactive Simulation
FDDI	Fiber Distributed Data Interface
FSM	Flight Station Module
OTW	Out-The-Window
SDD	Software Design Document
SSM	Simulation System Module
SSS	System/Segment Specification
TCP/IP	Transmission Communications Protocol / Internet Protocol
VSC	Visual System Controller
VSM	Visual System Module

**9 APPENDICES**

There are no Appendixes to this Software Design Document.